EMBEDDED
LINUX
CONFERENCE

ELC-E Dublin 2022

Slides: **tinyurl.com/elce22-bof**

Baylibre

RISC-V and Open Hardware BoF

Drew Fustini <dfustini@baylibre.com>

RISC-V: a Free and Open ISA

- Started by a computer architecture research group at University of California Berkeley in 2010 led by Krste Asanovic
- **V** as in the roman numeral five, because it is the 5th **RISC** instruction set to come out of UC Berkeley
- **Free and Open** because the specifications are published under an open source license: Creative Commons Attribution 4.0 International
  - Volume 1, Unprivileged Spec v. 20191213 [PDF]
  - Volume 2, Privileged Spec v. 20211203 [PDF]

What is different about RISC-V?

- Simple clean-slate design
  - Avoids any dependencies on microarchitecture style *(in-order, out-of-order, etc)*
- Modular design
  - Suitable for everything from microcontrollers to supercomputers
- Stable base
  - Base integer ISAs and standard extensions are frozen
  - Additions via optional extensions, not new versions

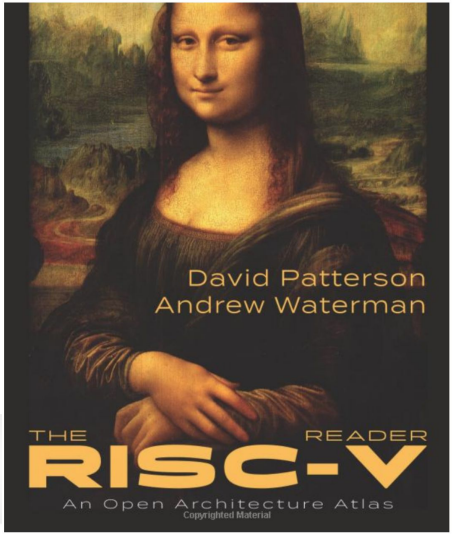
(source: Instruction Sets Want to be Free, Krste Asanovic)

Ratified in 2021

- 15 new specifications representing more than 40 extensions
- Vector
- Hypervisor
- Scalar Cryptography
- Bit Manipulation

Learn more about RISC-V

- Get up-to-speed quick with the RISC-V Reader



RISC-V International

- The organization that develops the RISC-V specifications: **riscv.org**
- Non-profit with 2,700+ members including companies and universities
- Become a member - *free of cost to individuals and non-profits*
- RISC-V wiki has many helpful resources
- RISC-V mailing lists for many topics *(public archives)*
- Technical Meetings Calendar

RISC-V Developer Boards

- Initiative from RISC-V International to get Linux-capable boards into the hands of open source developers
  - Launched in 2021 with the Allwinner D1 Nezha and SiFive Unmatched
- Fill out this form to apply
  - Need to be RISC-V International member (or part of a member organization), but remember that individuals can join RISC-V International free of cost
  - Explain why you are interested in RISC-V and what you plan to do with dev board
  - To improve your chances, don't overestimate your hardware requirements like RAM

Recent RISC-V presentations

- Linux on RISC-V *(Earlier today at ELC-E 2022 Dublin)*
- RISC-V micro-conference at Linux Plumbers Conference *(Sept 12)*
  - Live stream *(individual videos of talks posted later)*
- RISC-V Spring Week 2022 Videos on the RISC-V YouTube channel
  - State of the Union & the Road Ahead
  - Maturing the RISC-V Ecosystem
  - Evolving the Role of Software in the RISC-V Ecosystem

RISC-V Open Hours

- Bi-weekly virtual meetup for the community to interact in real-time
  - Primary focus on RISC-V support in open source software and RISC-V dev boards
  - Call for participation is open! No prepared talk or slides required!
- Schedule
  - Wednesday, October 12, 7:00 PM (US PDT) which is Thursday morning in Asia
  - Wednesday, October 26, 9:00 AM (US PDT) which is early evening in Europe

“Is RISC-V an Open Source processor?”

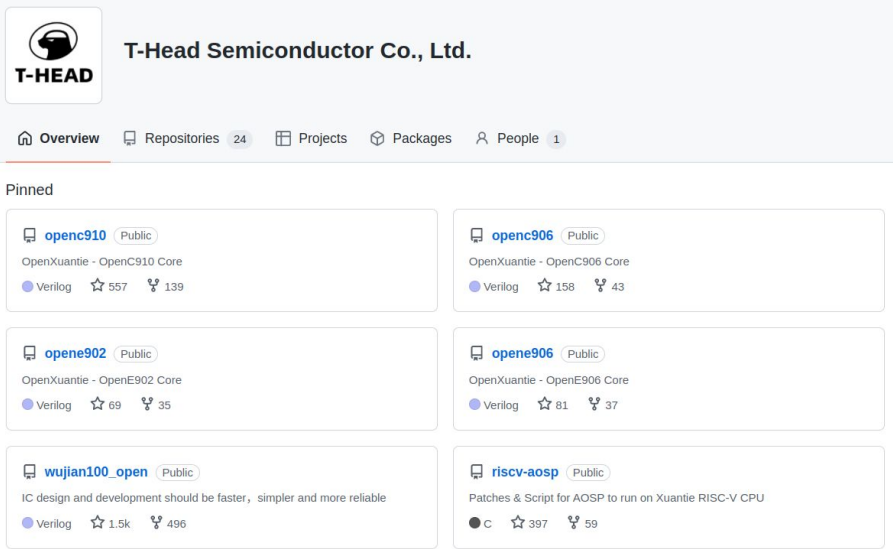
- RISC-V is a set of specifications under an open source license
- RISC-V implementations can be open source or proprietary
- Open specifications make open source implementations possible
- **An open ISA makes it possible to design an open source processor**

RISC-V open source cores

- Academia
  - Rocket and BOOM from Berkeley, PULP family of cores from ETH Zurich
- Industry
  - SweRV created by Western Digital and now developed by CHIPS Alliance
  - OpenHW Group creating proven verified IP like their Core-V designs
  - Google OpenTitan silicon root of trust project uses LowRISC Ibex core
- FPGA soft-cores
  - PicoRV32, RVfpga, SERV, and VexRiscV

Alibaba T-Head open source RISC-V cores

- OpenE902, OpenE906, OpenC906 (used in Allwinner D1 SoC), and OpenC910 cores on GitHub under permissive Apache 2.0 licence




```
|--C910_RTL_FACTORY/
  |--gen_rtl/     ## Verilog source code of C910
  |--setup/       ## Script to set the environment variables
|--smart_run/     ## RTL simulation environment
  |--impl/        ## SDC file, scripts and file lists for implementation
  |--logical/     ## SoC demo and test bench to run the simulation
  |--setup/       ## GNU tool chain setting
  |--tests/       ## Test driver and test cases
  |--work/        ## Working directory for builds
  |--Makefile     ## Makefile for building and running sim targets
|--doc/           ## The user and integration manual of C910
```

XiangShan (香山)

- open source high-performance RISC-V processor project from the Chinese Academy of Science
- RISC-V Summit 2021 talk by Professor Yungang Bao (slides)
  - "Contribute to XiangShan and realize your ideas on real chips! The open-source XiangShan will be taped-out every ~6 months"
- Nanhu is the 2nd generation microarchitecture
  - Target: 2GHz@14nm, SPEC CPU2006 20 marks; 7.81 CoreMark/MHz

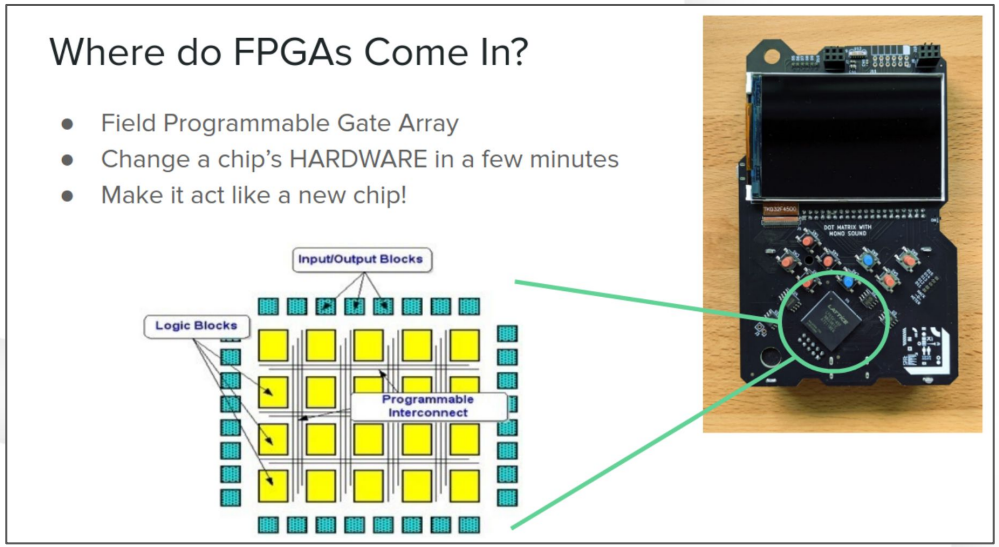
Open source silicon

- FOSSi Foundation (Free and Open Source Silicon)
  - El Correo Libre monthly newsletter for the latest on open source cores
- Build your own open source SoC with an open source silicon toolchain
  - Google and eFabless worked with Skywater to open source their 130 nm PDK (process design kit) and more recently 90 nm process too.
  - Google offers free-of-cost MPW (multi-project wafer) runs to open source projects
  - Learn to design your own using open source design tools with Zero to ASIC course.

BONUS:
What about RISC-V on FPGAs?

Introduction

- "RISC-V and FPGAs: Open Source Hardware Hacking" keynote at Hackaday Supercon 2019 by Dr. Megan Wachs



Open source FPGA toolchains

- Project IceStorm for Lattice iCE40 FPGA
  - "A Free and Open Source Verilog-to-Bitstream Flow for iCE40 FPGAs" by Claire Wolf
- Project Trellis for the more capable Lattice ECP5 FPGA
  - "Project Trellis and nextpnr FOSS FPGA flow for the Lattice ECP5" by Myrtle Shah
- Project X-Ray and SymbiFlow for much more capable Xilinix Series 7
  - "Xilinx Series 7 FPGAs Now Have a Fully Open Source Toolchain!" by Tim Ansell
  - "Open Source Verilog-to-Bitstream FPGA synthesis flow, currently targeting Xilinx 7-Series, Lattice iCE40 and Lattice ECP5 FPGAs. Think of it as the GCC of FPGAs"

Why design an SoC in Python?

- Python has advantages over traditional HDL like VHDL and Verilog
  - Many people are familiar with Python than HDL (hardware description languages)
  - There are currently more software developers than hardware designers
- Migen is a Python framework that can automate chip design
  - Leverages the object-oriented, modular nature of Python
  - Produces Verilog code so it can be used with existing chip design workflows
- "Using Python for creating hardware to record FOSS conferences!"

What is Migen?

migen BASICS

```vhdl
-- Libraries imports
library ieee;
use ieee.std_logic_1164.all;

-- Module interface description
entity my_module is
    port(
        clk : in std_logic;
        o   : out std_logic
    );
end entity;

-- Module architecture description
architecture rtl of my_module is
   signal d : std_logic;
   signal q : std_logic;
begin
    -- Combinatorial logic
    o <= q;
    d <= not q;

    -- Synchronous logic
    process(clk)
    begin
        if rising_edge(clk) then
            d <= q
        end if;
    end process

end rtl;
```

VHDL

An alternative HDL based on Python

```python
from migen import *

class MyModule(Module):
    def __init__(self):
        self.o = Signal()

        # # #

        d = Signal()
        q = Signal()

        # combinatorial logic
        self.comb += [
            self.o.eq(q),
            d.eq(~q)
        ]

        # synchronous logic
        self.sync += d.eq(q)
```

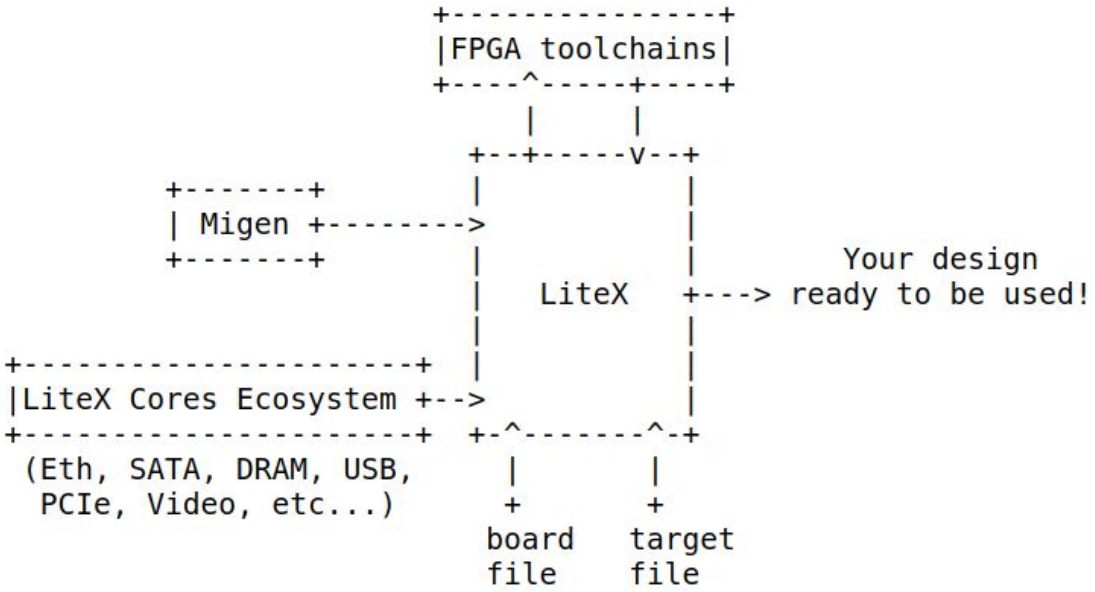
Migen

source: http://goo.gl/mZJvFQ

EnjoyDigital

LiteX

- Based on Migen, builds full SoC that can be loaded into an FPGA



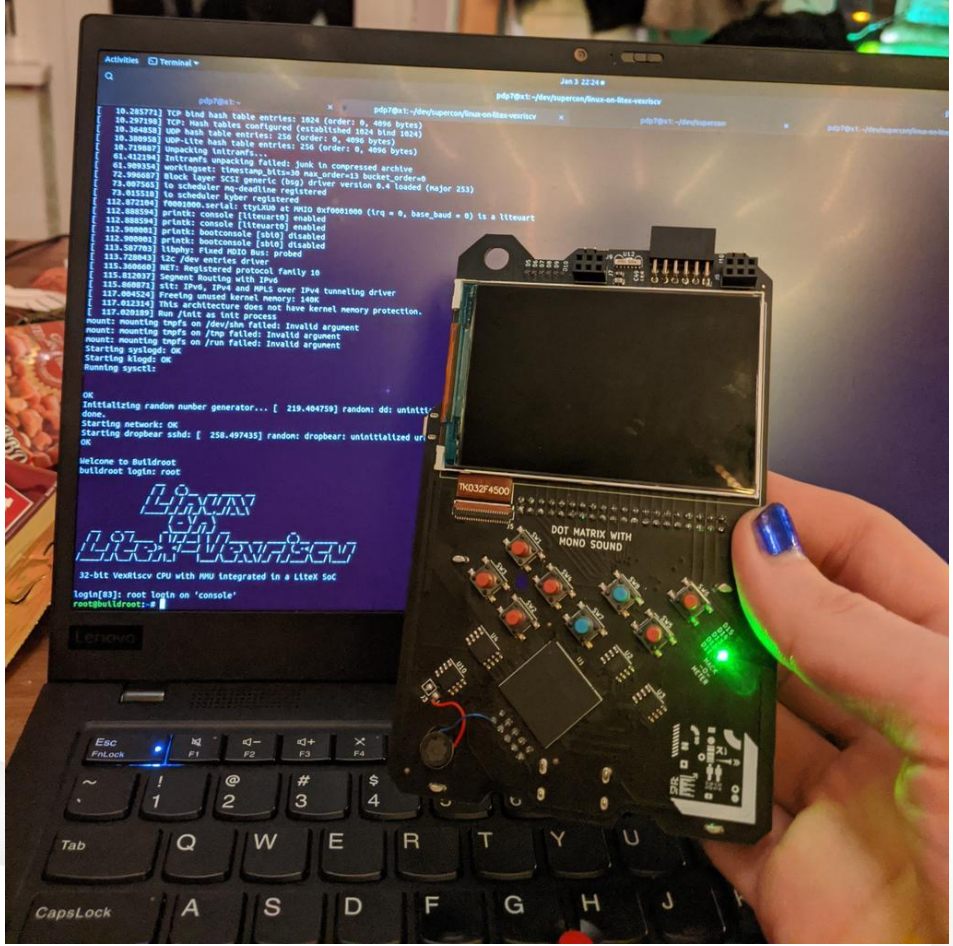
LiteX

- "LiteX vs. Vivado: First Impressions"
- Collection of open cores for DRAM, Ethernet, PCIe, SATA and more...

Name	Build Status	Description
LiteDRAM	build passing	DRAM
LiteEth	build passing	Ethernet
LitePCIe	build passing	PCIe
LiteSATA	build passing	SATA
LiteSDCard	build passing	SD card
LiteICLink	build passing	Inter-Chip communication
LiteJESD204B	build passing	JESD204B
LiteVideo	build unknown	VGA, DVI, HDMI
LiteScope	build passing	Logic analyzer

Linux on LiteX-VexRiscv

- VexRiscv: 32-bit Linux-capable RISC-V core
  - Designed to be FPGA friendly
  - Written in Spinal HDL (based on Scala)
- Builds an SoC using VexRiscv core and LiteX modules
  - Such as LiteDRAM, LiteEth, LiteSDCard, LitePCIe
  - “This project demonstrates how high level HDLs (Spinal HDL, Migen) enable new possibilities and complement each other. Results shown here are the results of a productive collaboration between open-source communities”
- Supports large number of FPGA dev boards including Digilent Arty A7


Activities
Terminal
Welcome to Buildroot
buildroot login: root
Linux on LiteX-VexRiscv
32-bit VexRiscv CPU with MMU integrated in a LiteX SoC
login[83]: root login on 'console'
root@buildroot:~#
TK032F4500
DOT MATRIX WITH MONO SOUND
Lenovo

litex-hub / **litex-boards**

Unwatch 7 | Unstar 17 | Fork 24

Code | Issues 2 | Pull requests 1 | Actions | Projects 0 | Wiki | Security | Insights

add the Hackaday Supercon ECP5 badge #31

Edit

Merged **enjoy-digital** merged 1 commit into `litex-hub:master` from `pdp7:master` 21 days ago

Conversation 18 | Commits 1 | Checks 1 | Files changed 2 | +461 −0

pdp7 commented 22 days ago • edited — Contributor

Add the Hackaday Supercon 2019 badge which has an ECP5 FPGA.

These changes are from a fork by Michael Welling (**@mwelling**)

During Supercon, we tried two approaches:

- use the built-in 16MB QSPI SRAM
- use add-on cartiridge with 32MB SDRAM by Jacob Creedon

We were not able to get the QSPI SRAM working so I've removed those changes, and I have just added the changes that are needed to boot Linux with the 32MB SDRAM.

In addition to **@mwelling**, thank you to Jacob Creedon (**@jcreedon**), **@gregdavill**, Tim Ansell (**@mithro**), and Sean Cross (**@xobs**) who all helped get Linux working on this badge.

KiCad design files by **@jcreedon** for the SDRAM cartridge are available on GitHub.

Reviewers
No reviews

Assignees
No one assigned

Labels
None yet

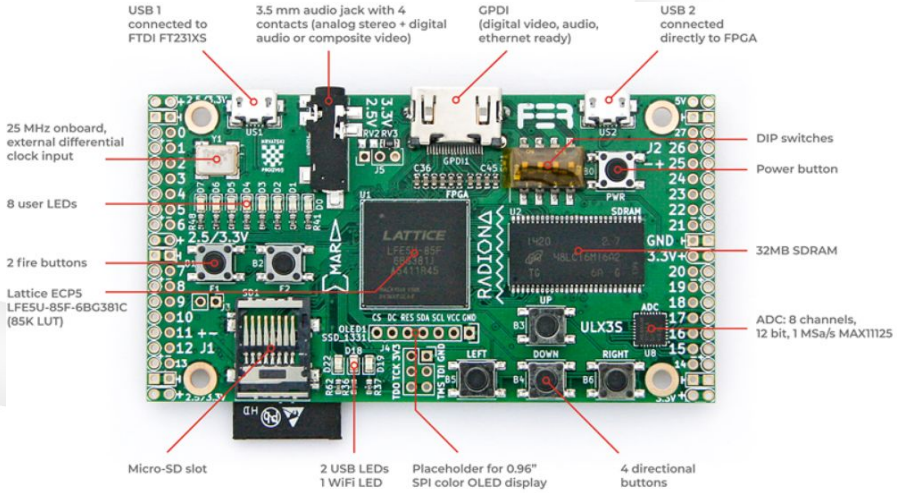
Projects
None yet

Milestone
No milestone

Open Source ECP5 FPGA boards

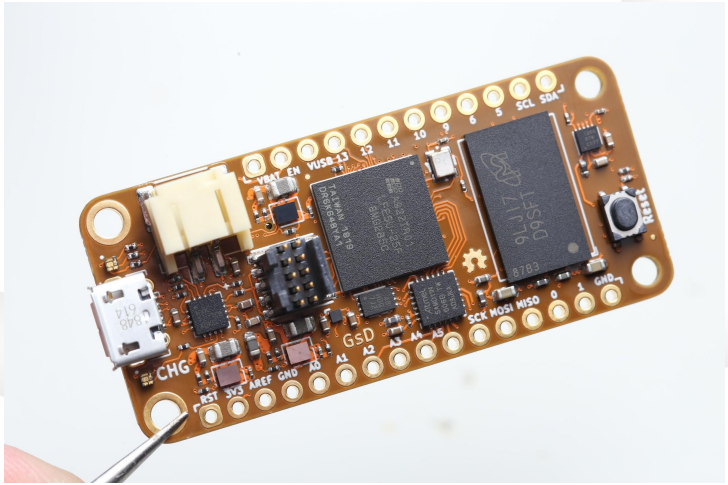
- [Radiona.org ULX3S]

  - 32MB SDRAM; ESP32 on-board for WiFi and Bluetooth; $115 on [CrowdSupply] or [Mouser]



Open Source ECP5 FPGA boards

- OrangeCrab by Greg Davill
  - 128MB DDR RAM; Adafruit Feather form factor; available for $129



Want to learn FPGAs? Try Fomu!

- Online workshop from Tim Ansell and Sean Cross
- $50 on CrowdSupply
- Fits inside USB port!
- Learn how to use:
  - MicroPython
  - Verilog
  - LiteX